

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1-52. (canceled)

53. (currently amended) A service provision system for use in distributed processing environments, said system comprising:
an input for receiving a service request;
processing means for processing the service request ;
negotiation means for use in establishing conditions applicable to provision of one or more component processes involved in provision of the service, said negotiation means being adapted to assemble said conditions ~~in the data store~~ proactively by negotiation prior to receipt of said service request;
an up-datable data store;
means to access said up-datable data store for storing said conditions when established and assembled; and
an output for providing a response to the service request, said response comprising an indication of availability of the requested service;
wherein the processing means is adapted to process a service request by accessing one or more of the previously established conditions in the data store, processing the request using the one or more established conditions and producing said response.

54. (previously presented) A service provision system as in claim 53

wherein one or more of said established conditions has an associated expiry time after which it is no longer applicable.

55. (previously presented) A service provision system as in claim 54

wherein the processing means is adapted to detect an expired or undefined condition in the data store, which condition is applicable to a component process for the provision of a requested service, and to trigger the negotiation means to establish a substitute condition.

56. (previously presented) A service provision system as in claim 55

further comprising:

means to access said data store for storing data related to services offered by the system and to one or more entities which have an interest in receiving information relating to one or more of said services, together with means to transmit information based on said data related to services to the one or more entities which have an interest.

57. (previously presented) A system as in claim 53 which further

comprises initiation means to initiate one or more component processes in provision of a requested service.

58. (previously presented) A service provision system as in claim 57

wherein:

provisioning a requested service requires provision of a selected set of component processes;

the negotiation means establishes and stores a set of conditions applicable to provision of the component processes of the selected set; and

the processing means is adapted to process a service request by accessing the stored set of conditions in the data store, processing the request using said stored set, and producing said response.

59-60. (canceled)

61. (previously presented) A service provision method for use in distributed processing environments, said method comprising:
establishing conditions applicable to provision of one or more component processes in a service, proactively by negotiation prior to receipt of said service request;
accessing an updatable data store and storing said conditions once established;
subsequently receiving a request for said service;
processing said service request by:
a) accessing one or more of said previously established conditions in the data store; and

b) providing a response to the service request, said response comprising an indication of availability of the requested service dependent upon whether said one or more established conditions is met.

62. (previously presented) A service provision method according to claim 61 wherein one or more of said established conditions stored in said data store is applicable until advent of an expiry time associated with said one or more conditions.

63. (previously presented) A service provision method according to claim 62 wherein further comprising the step, responsive to receipt of said request, of finding whether any conditions for provision of said service are extant and substituting a substitute condition in the event that no such conditions are found.

64. (previously presented) A service provision method according to claim 61 wherein said method further comprises the step of scheduling provision of said one or more component processes, said step being carried out after receipt of said request for said service.

65. (previously presented) A service provision method according to claim 64 wherein said method further comprises the step, responsive to a failure to schedule one or more component processes, of carrying out one of the following steps:

- i) re-schedule the component process;
- ii) transmit a message to an entity which requested the service indicating that the service can only be provided under conditions different to said previously established conditions;
- iii) re-assign the service to another service provider; or
- iv) indicate to an entity which requested the service that the requested service cannot be provided.

66. (previously presented) A method according to claim 61 further comprising the step of identifying component processes for use in provisioning the requested service.

67. (previously presented) A method according to claim 66 which further comprises initiating one or more of said component processes identified for use in the requested service.

68. (currently amended) A method of operating one or more computers to manage a business process, said method comprising:

executing a first autonomous software process ~~representing a service requester~~;

executing a second autonomous software process ~~representing a service provider~~;

the execution of said autonomous software processes establishing conditions applicable to provision of said service by negotiation between said first and second autonomous software processes;

accessing an updatable data store and storing said conditions in said data store once established;

the execution of said first autonomous software process subsequently involving the transmission receipt of a request for said service for handling by said second first autonomous software process; and

responsive to receipt of said request, said second first autonomous software process testing whether the established conditions for provision of said service are met and providing an indication as to whether the requested service is available in dependence upon the outcome of said test.

69. (previously presented) A method according to claim 68 in which one or more of said established conditions is stored in said data store until advent of an expiry time associated with said one or more conditions.

70. (currently amended) A method according to claim 68 wherein, responsive to receipt of said request, said second first autonomous software process tests whether any conditions for provision of said service are extant and substitutes a substitute condition in the event that no such conditions are found.

71. (currently amended) A method according to claim 69 wherein,

responsive to receipt of said request, said ~~second~~first autonomous software process tests whether any conditions for provision of said service are extant and substitutes a substitute condition in the event that no such conditions are found.

72. (currently amended) A method according to claim 68 in which said

~~second~~first autonomous software process identifies one or more further autonomous software processes representing resources required to provide one or more component services in the provision of said service, said method further comprising executing said further autonomous software process representing a component service provider.

73. (currently amended) A method according to claim 69 in which said

~~second~~first autonomous software process identifies one or more further autonomous software processes representing resources required to provide one or more component services in the provision of said service, said method further comprising executing said further autonomous software process representing a component service provider.

74. (currently amended) An apparatus for representing a provider of a

service component of a business process, said apparatus comprising:

a computer having ~~installed thereon~~ a memory storing autonomous software code executable to provide an autonomous software process representing a service requester,

the execution of said autonomous software process:

a) establishing conditions applicable to provision of said service by

negotiation between said autonomous software process and one or more other autonomous software processes;

b) accessing an up-datable data store and storing said conditions in said data

store once established; and

c) on subsequently receiving a request for said service, accessing said data

store and testing whether the established conditions for provision of said service are met and providing an indication as to whether the requested service is available in dependence upon the outcome of said test.

75. (previously presented) Apparatus according to claim 72 in which said autonomous software code is executable to provide a plurality of software threads, able to concurrently interact with different other autonomous software processes.

76. (new) A method of operating one or more computers to manage a business process, said method comprising:

executing a first autonomous software process;

executing a second autonomous software process;

the execution of said autonomous software processes establishing conditions applicable to provision of said service by negotiation between said first and second autonomous software processes;

accessing an updatable data store and storing said conditions in said data store once established;

the execution of said first autonomous software process subsequently involving the receipt of a request for said service for handling by said first autonomous software process; and

responsive to receipt of said request, said first autonomous software process testing whether the established conditions for provision of said service are met and providing an indication as to whether the requested service is available in dependence upon the outcome of said test;

wherein there are no control dependencies between the first and second autonomous software processes.